

O'Reilly News: XML

<http://oreilly.com/xml/>

An SOE comedy of errors

Over the years, the IT section has been pro-actively trying to improve security, to lock the system down, and keep commercial information safe. However, these steps also prevent the engineers from getting their jobs done, so they are all circumvented. The more that the official, central SOE is locked down, the more that the remote users have banded together to make their own unofficial SOE that doesn't get in the way.

<http://feeds.oreilly.com/~r/oreilly/xml/~3/pyN1BR095UM/an-soe-comedy-of-errors.html>

What's New in O'Reilly Answers

[Best screencasting software](#)[Which Twitter client is best for Android phones?](#)[Should I buy an iPad?](#)[How to clean up after a tarball or zipball explodes](#)[Macros in XSLT](#) Share knowledge, ask questions on [O'Reilly Answers](#) today.

<http://feeds.oreilly.com/~r/oreilly/xml/~3/iWtkQBn9Dns/>

Japanese Standard for ODF

Based on a [cryptic twitter](#) from Dr Murata, it looks like the Japanese standard for ODF has been released. Congratulations to all involved, it is a good step forward to enable competition, substitution and industry in this area.

http://feeds.oreilly.com/~r/oreilly/xml/~3/j_HtFD4WysU/japanese-standard-for-odf.html

Climategate and XML

One interesting artifact to come out of the stolen Climategate material is an epic file HARRY_READ_ME.txt. It seems to be a year long log by a programmer (Harry?) who has to port old data and various old FORTRAN (and MATLAB?)...

<http://feeds.oreilly.com/~r/oreilly/xml/~3/5mt7NiY1R8g/climategate-and-xml.html>

Vale JCP?

I am really impressed by Scala, though I have not used it on any real projects yet. Apart from reflection, it seems to be much stronger than Java in all the kinds of features that are good for XML document processing: co-routines, pattern matching and so on. The built-in XML tree that documents can be parsed in to does not contain back pointers, so up-going axes require extra coding; Scala is obviously more congenial for OmniMark or XSLT programmers than Java.

<http://feeds.oreilly.com/~r/oreilly/xml/~3/m0EDcEJ8AQg/vale-jcp.html>

How far can documentation go?

SAMBA's [Jeremy Allison](#) has a great post [Why writing a Windows compatible file server is \(still\) hard](#). What leaps out to me? First, that the method of requiring complete documentation outside a formalized QA process doesn't work real well. The second thing is that even if there is documentation, some incompatibilities come down to capability mismatches.

http://feeds.oreilly.com/~r/oreilly/xml/~3/7BfxUPb_ew/how-far-can-documentation-go.html

Schematron at the Associated Press

Stuart Myles has a quick slide presentation [Schematron and Other Useful Tools](#) at the [IPTC](#) Autumn Meeting about how the Associated Press reduced manual checking & QA of incoming iAtom feeds using open source tools.

<http://feeds.oreilly.com/~r/oreilly/xml/~3/32qtY586Kro/schematron-at-the-associated-p.html>

How fuzzy should a date be?

From Bruce D'Arcus' [Darcusblog](#) comes a pointer on a U.S. Library of Congress initiative for a better date format [Extended Date Time Format](#) (EDTF). ISO 8601's problem is that almost anything is a date: if my memory serves me, some date values are ambiguous so you need to make a subset or add some attribute to say which kind of date you mean.

http://feeds.oreilly.com/~r/oreilly/xml/~3/yLfdt32h_EA/how-fuzzy-should-a-date-be.html

Leaked Draft of EU Interop Framework

Two months ago I alerted readers [Europeans: only two weeks left to comment on ICT & standards whitepaper](#). I am not sure on which dots actually join up, but a Dutch website has what is claimed to be a leaked late draft in English of [European Interoperability Framework for European Public Services \(EIF\) Version 2.0](#). Here are some of the general recommendations related to standards and issues raised on this blog.

<http://feeds.oreilly.com/~r/oreilly/xml/~3/T3gNYCctjWE/leaked-draft-of-eu-interop-fra.html>
